joynr
April 2018 | distributed applications made easy

Manuel Schiller
BMW Car IT GmbH
Agenda

• Developing a connected feature
• Technical overview of joynr
• joynr use cases @ BMW
• joynr’s development model
Developing a connected feature
Sample Application: TrafficInfo

Requirement 1: Communication

subscribe to location A
subscribe to location A
subscribe to location B
Sample Application: TrafficInfo

Requirement 1: Communication
Sample Application: TrafficInfo

Requirement 1: Communication
MQTT is a publish-subscribe-based "lightweight" messaging protocol for use on top of the TCP/IP protocol.
MQTT
Sample Application: TrafficInfo

Requirement 2: Interface and Datatypes

geocast:

- subscribe to sampleapplication/TRAFFIC to receive traffic updates
- subscribe to sampleapplication/ACCIDENT to receive accident overlays

ACCIDENT contains:

- location
- numberOfLanesBlocked
- Direction

TRAFFIC contains...
modelling

MQTT
Sample Application: TrafficInfo

Requirement 3: Send commands and associate responses

trigger topics:
- trigger is sent on sampleapplication/$VIN containing triggerId
- response is sent on sampleapplication/responses with triggerId

vehicle sends detected accidents:
- detected accident is sent on sampleapplication/accidentdetected
- response is sent on sampleapplication/$VIN/accidentdetected
Sample Application: TrafficInfo

Requirement 4: Data serialization
modelling
RPC
serialization

MQTT
Sample Application: TrafficInfo

Requirement 5: Interface versioning

ACCIDENT events extended to support autonomous driving:

- blockedShape: raster outlines the areas of lanes affected

geocast:

- subscribe to sampleapplication/TRAFFIC to receive traffic updates
- subscribe to sampleapplication/v2/ACCIDENT to receive accident overlays
MQTT

- versioning
- modelling
- RPC
- serialization
Sample Application: TrafficInfo

Requirement 6: Security

Message integrity must be guaranteed to prevent spoofed accident warnings from being disseminated in the system.
Sample Application: TrafficInfo

Requirement 6: Privacy

Data privacy law in effect, VIN cannot be longer used in MQTT topic.

Solution: Adding a layer of indirection.
Resulting stack

- For this specific use case
- Other applications have similar requirements
Our generalized solution

joynr is a fault-tolerant, typed communications-middleware abstraction framework for applications and services deployed to vehicles, consumer devices and backend servers that need to interact with each other.
Technical overview of joynr
High level architecture – Global communication

The joynr network is structured in interconnected clusters:

- Cluster 1
- Cluster 2
- Cluster 3
- Cluster 4
- Cluster 5

Central message broker

Global Capabilities Directory
High level architecture – local communication

- each cluster is managed by a cluster-controller
- maintains the connectivity to the outside world
- acts as a message broker
- provides a discovery service for nodes connected to it
High level architecture – local communication

• each node is
  – either a provider implementing an interface
  – or a proxy accessing an interface.

• multiple providers can implement the same interface within a cluster
Message routing

• joynr is an overlay network
• each node has its globally unique identifier
Technical details

• implementations currently exist in
  – C++
  – Java / JEE
  – JavaScript

• based on Franca IDL

• currently supports WebSockets, HTTP and MQTT as transport layers
joynr use cases @ BMW
Deployment

vehicle 1

ECU

vehicle n

ECU

MQTT broker

backend application 1

backend application 2

backend application k

Global Capabilities Directory
joynr use cases @ BMW

• Unidirectional uploading of data

• Dissemination of information to a multitude of receivers
  – Multicasting
  – Geocasting

• RPC in both directions
  – Trigger actions on vehicle remotely
  – Access backend data on demand from vehicle

• intra-ECU communication
joynr’s development model
Development model

- joynr is an open source project: https://github.com/bmwcarit/joynr

- licensed under Apache 2.0

- development and maintenance by BMW Car IT GmbH

- contributions and pull requests are welcome!
Thank you!

Visit joynr at http://joynr.io
Contact us: joynr@bmw-carit.de